

29장 Math

- 표준 빌트인 객체인 Math는 수학적인 상수와 함수를 위한 프로퍼티와 메서드를 제공한다.
- Math는 생성자 함수가 아니다. 따라서 정적 프로퍼티와 정적 메서드만 제공한다.

29.1 Math 프로퍼티

29.1.1 Math.PI

- 원주율 PI값을 반환한다.

```
Math.PI; // -> 3.141592653589793
```

29.2 Math 메서드

29.2.1 Math.abs

- Math.abs메서드는 인수로 전달된 숫자의 절대값을 반환한다.
- 절대값은 반드시 0 또는 양수여야한다.

```
Math.abs(-1);           // -> 1
Math.abs('-1');         // -> 1
Math.abs('');           // -> 0
Math.abs([]);           // -> 0
Math.abs(null);         // -> 0
Math.abs(undefined);   // -> NaN
Math.abs({});           // -> NaN
Math.abs('string');    // -> NaN
Math.abs();             // -> NaN
```

29.2.2 Math.round

- Math.round 메서드는 인수로 전달된 숫자의 소수점 이하를 반올림한 정수를 반환한다.

```
Math.round(1.4); // -> 1
Math.round(1.6); // -> 2
Math.round(-1.4); // -> -1
Math.round(-1.6); // -> -2
Math.round(1); // -> 1
Math.round(); // -> NaN
```

29.2.3 Math.ceil

- Math.ceil 메서드는 인수로 전달된 숫자의 소수점 이하를 올림한 정수를 반환한다.

```
Math.ceil(1.4); // -> 2
Math.ceil(1.6); // -> 2
Math.ceil(-1.4); // -> -1
Math.ceil(-1.6); // -> -1
Math.ceil(1); // -> 1
Math.ceil(); // -> NaN
```

29.2.4 Math.floor

- Math.floor 메서드는 인수로 전달될 숫자의 소수점 이하를 내림한 정수를 반환한다.

```
Math.floor(1.9); // -> 1
Math.floor(9.1); // -> 9
Math.floor(-1.9); // -> -2
Math.floor(-9.1); // -> -10
Math.floor(1); // -> 1
Math.floor(); // -> NaN
```

29.2.5 Math.sqrt

- Math.sqrt 메서드는 인수로 전달된 숫자의 제곱근을 반환한다.

```
Math.sqrt(9); // -> 3
Math.sqrt(-9); // -> NaN
```

```
Math.sqrt(2); // -> 1.414213562373095
Math.sqrt(1); // -> 1
Math.sqrt(0); // -> 0
Math.sqrt(); // -> NaN
```

29.2.6 Math.random

- Math.random 메서드는 임의의 난수(랜덤 숫자)를 반환한다.
- Math.random 메서드가 반환한 난수는 0에서 1 미만의 실수다. 즉, 0은 포함되지만 1은 포함되지 않는다.

```
Math.random(); // 0에서 1 미만의 랜덤 실수(0.8208720231391746)

/*
1에서 10 범위의 랜덤 정수 취득
1) Math.random으로 0에서 1 미만의 랜덤 실수를 구한 다음, 10을 곱해
0에서 10 미만의
랜덤 실수를 구한다.
2) 0에서 10 미만의 랜덤 실수에 1을 더해 1에서 10 범위의 랜덤 실수를
구한다.
3) Math.floor로 1에서 10 범위의 랜덤 실수의 소수점 이하를 떼어 버린
다음 정수를 반환한다.
*/
const random = Math.floor((Math.random() * 10) + 1);
console.log(random); // 1에서 10 범위의 정수
```

29.2.7 Math.pow

- Math.pow 메서드는 첫 번째 인수를 밑으로 두번째 인수를 지수로 거듭제곱한 결과를 반환한다.

```
Math.pow(2, 8); // -> 256
Math.pow(2, -1); // -> 0.5
Math.pow(2); // -> NaN
```

- Math.pow 메서드 대신 ES7에서 도입된 지수 연산자를 사용하면 가독성이 좋다.

```
// ES7 지수 연산자
2 ** 2 ** 2; // -> 16
Math.pow(Math.pow(2, 2), 2); // -> 16
```

29.2.8 Math.max

- Math.max 메서드는 전달받은 인수 중에서 가장 큰 수를 반환한다.
- 인수가 전달되지 않으면 -Infinity를 반환한다.

```
Math.max(1); // -> 1
Math.max(1, 2); // -> 2
Math.max(1, 2, 3); // -> 3
Math.max(); // -> -Infinity
```

- 배열을 인수로 전달받아 배열의 요소 중에서 최대값을 구하려면 Function.prototype.apply 메서드 또는 스프레드 문법을 사용해야한다.

```
//배열 요소 중에서 최대값 취득
Math.max.apply(null, [1, 2, 3]); // -> 3

// ES6 스프레드 문법
Math.max(...[1, 2, 3]); // -> 3
```

29.2.9 Math.min

- Math.min 메서드는 전달받은 인수 중에서 가장 작은 수를 반환한다.
- 인수가 전달되지 않으면 Infinity를 반환한다.

```
Math.min(1); // -> 1
Math.min(1, 2); // -> 1
Math.min(1, 2, 3); // -> 1
Math.min(); // -> Infinity
```

- 배열을 인수로 전달받아 배열의 요소 중에서 최소값을 구하려면 Function.prototype.apply 메서드 또는 스프레드 문법을 사용해야한다.

```
//배열 요소 중에서 최소값 취득  
Math.min.apply(null, [1, 2, 3]); // -> 1  
  
// ES6 스프레드 문법  
Math.min(...[1, 2, 3]); // -> 1
```